

RL

The opinion in support of the decision being entered today was not written for publication and is not binding precedent of the Board.

Paper No. 20

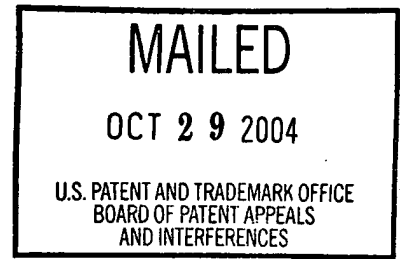
UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte BRYAN KEITH BULLIS,
RAJ KUMAR SINGH,
and FOSTER BEAVER WHITE

Appeal No. 2004-1640
Application 09/409,940¹

ON BRIEF



Before HAIRSTON, JERRY SMITH, and BARRETT, Administrative Patent Judges.

BARRETT, Administrative Patent Judge.

DECISION ON APPEAL

This is a decision on appeal under 35 U.S.C. § 134 from the final rejection of claims 1-23.

We affirm.

¹ Application for patent filed September 30, 1999, entitled "Method and System for Providing Hierarchical Self-Checking in ASIC Simulation."

BACKGROUND

The invention relates to a system and method for providing simulation of an integrated circuit.

Claim 1 is reproduced below.

1. A system for providing simulation of an integrated circuit during development of the integrated circuit, the integrated circuit having an island including an interface, the system comprising:

a snooper coupled with the interface for monitoring the interface and obtaining an output provided by the island during simulation;

a checker, coupled with the interface, for checking the output to determine whether the output is a desired output;

a generator coupled with the island for providing an input to the island during simulation; and

at least one test case coupled with the generator for directing the generator;

wherein the checker further determines the desired output based upon the input; and

wherein the generator further includes intelligence to provide the input to the island based only upon data and a request provided by the at least one test case to the generator, the request requesting that the generator perform a particular simulation on the island.

THE REFERENCES

The examiner relies on the following references:

Guruswamy et al. (Guruswamy)	6,006,024	December 21, 1999 (filed November 1, 1996)
Hollander	6,182,258	January 30, 2001 (filed February 6, 1998)

THE REJECTIONS

Claims 1-23 stand rejected under 35 U.S.C. § 112, first paragraph, based on lack of enablement.

Claims 1-23 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Hollander and Guruswamy.

We refer to the final rejection (Paper No. 7) (pages referred to as "FR__") and the examiner's answer (Paper No. 15) (pages referred to as "EA__") for a statement of the examiner's rejection, and to the brief (Paper No. 14) (pages referred to as "Br__") and reply brief (Paper No. 16) (pages referred to as "RBr__") for a statement of appellants' arguments thereagainst.

OPINION

Enablement

The examiner concludes that the specification does not describe, in a way that would allow one skilled in the art to make and/or use it, the "snooper," the "interface," the "checker," the "generator," and the "test case" in system claims 1-9, the corresponding steps of "snooping the interface," "checking the output," providing input using a "generator," and directing the providing of input using a "test case" in method claims 10-16, and corresponding limitations in the computer readable medium having a program of instructions in claims 17-23.

Appellants argue that the examiner has completely failed to explain why the specification is not enabling. It is argued that

Appeal No. 2004-1640
Application 09/409,940

one of ordinary skill in the art would understand the terms interface, snooper, checker, generator, and how the test case directs the generator and how to make and use the invention (Br8-15). Appellants have provided a first declaration (Paper No. 5) and a second declaration (Paper No. 10) by co-inventor Raj Singh (Paper No. 5).

The examiner responds that the specification does not contain any specifics (EA12-15).

Appellants respond at length to the examiner's statements (RBr3-8).

"The test of enablement is whether one reasonably skilled in the art could make or use the invention from the disclosures in the patent coupled with information known in the art without undue experimentation." United States v. Telectronics, Inc., 857 F.2d 778, 785, 8 USPQ2d 1217, 1223 (Fed. Cir. 1988). A patent need not teach, and preferably omits, what is well known in the art. Paperless Accounting, Inc. v. Bay Area Rapid Transit System, 804 F.2d 659, 664, 231 USPQ 649, 652 (Fed. Cir. 1986). The USPTO must support a rejection for lack of enablement with reasons. In re Marzocchi, 439 F.2d 220, 223-24, 169 USPQ 367, 369-70 (CCPA 1971). The factors to be considered in determining whether a disclosure would require "undue experimentation" are summarized in In re Wands, 858 F.2d 731, 737, 8 USPQ2d 1400, 1404 (Fed. Cir. 1988): (1) the quantity of experimentation necessary;

(2) the amount of direction or guidance presented; (3) the presence or absence of working examples; (4) the nature of the invention; (5) the state of the prior art; (6) the relative skill of those in the art; (7) the predictability or unpredictability of the art; and (8) the breadth of the claim. See also MPEP § 2164.01(a) (8th ed., Rev. 1, Feb. 2003). The Wands factors "are illustrative, not mandatory. What is relevant depends on the facts." Amgen, Inc. v. Chugai Pharm. Co., Ltd., 927 F.2d 1200, 1213, 18 USPQ2d 1016, 1027 (Fed. Cir. 1991).

It appears that the enablement rejection is based on the lack of specific detailed descriptions of circuits and programs for the contested terms. However, what is important is whether the specification would be enabling to one of ordinary skill in the art. The examiner does not address any of the Wands factors. We consider the three factors of the amount of direction or guidance presented, the state of the prior art, and the relative skill of those in the art. The specification, and indeed claim 1, contains a description of the function of the interface, snooper, checker, generator, and how the test case directs the generator. Thus, the specification provides guidance on what functions are to be performed and how the various functions are interconnected. The level of ordinary skill in the relevant art of integrated circuit verification appears to be very high and, thus, things that are well known do not have to be expressly

disclosed. It is the examiner's responsibility to establish why one having a presumed high level of skill in the art could not make the invention. For example, "a snooper coupled with the interface for monitoring the interface and obtaining an output provided by the island during simulation" simply requires receiving an output via the interface. Since it is so common in electrical and computer engineering to receive an output it is not at all apparent to us why undue experimentation would be required. Similarly, providing an input and checking the output to determine whether the output matches a desired output seem to be common electrical and computer operations. The state of the prior art also shows that there is no enablement problem. For example, Fig. 1 of Hollander shows nothing more than a line providing an input and a line providing an output and box for a generator and a box for a checker. If this disclosure is enabling, (and, being a patent, it must be presumed to be enabling), then appellants' disclosure, which is of a comparable scope, must also be enabling or, at least, the examiner has provided no reasons why Hollander is enabled but why the claimed invention is not enabled. It is not necessary to address the first and second declarations. We conclude that the examiner has failed to establish a prima facie case of lack of enablement. The rejection of claims 1-23 is reversed.

Obviousness

The rejection and arguments

The examiner finds that Hollander discloses the claimed invention except for testing an "island" (EA10). In particular, the examiner refers to the abstract; col. 3, line 37; col. 4, line 66 to col. 5, line 7; col. 8, line 30; col. 10, line 21; and Figs. 1-5. The examiner finds that Guruswamy discloses a cell layout generation that includes islands for integrated circuit design (EA10) and concludes that it would have been obvious to one of ordinary skill in the art to apply the verification system of Hollander to islands in view of Guruswamy (EA10-11).

Appellants argue that in the claimed invention "the checker can both generate desired outputs and check the outputs from the island under test against the desired inputs [sic, outputs]" (Br16; RBr8-9) and "Hollander in view of Guruswamy fails to teach or suggest the use of a checker that both generates the desired inputs [sic, outputs] and checks the actual inputs [sic, outputs] against the desired inputs [sic, outputs]" (Br17). It is argued that appellants find "no mention in the cited portion of Hollander that synchronizing could or should include using the checking module to generate the desired outputs based upon the inputs from the test generation module" (Br17) and "Applicant can find no indication that it is the checking module, not the test

generation module or some other module, that determines the desired outputs" (Br17; RBr9).

The examiner does not answer these arguments. The examiner states that appellants appear to argue that Hollander both teaches synchronizing the checking module and does not teach synchronizing the checking module (EA15). It is stated that this does not make sense, but, in any case, synchronization is not recited in the claims (EA15). The examiner fails to appreciate that the rejection relies on the synchronization at column 8, line 30 of Hollander and that appellants are trying to address this disclosure. The examiner also states that appellants argue that Guruswamy does not teach checking the behavior of a circuit, but that Hollander is relied on for this feature (EA15-16). In both cases, the examiner refers to statements made by appellants in the description of the prior art, not in the argument section of the brief, and the examiner disregards appellants' arguments that Hollander and Guruswamy do not teach using a checking module or checker "to generate the desired outputs based upon the inputs" (Br7, two places).

Appellants respond that Hollander teaches synchronizing the checking module, but disagrees that "this synchronizing is synonymous with using the checker to both generate desired outputs and check the outputs from the island under test against the desired inputs [sic, outputs]" (RBr9). Appellants further

argue that they "can find no mention in the cited portions of Guruswamy of using a checker to not only check the outputs of the island under test, but also to generate the desired outputs based upon the inputs" (RBr9).

Analysis

Appellants argue independent claims 1, 10, and 17 together.

Appellants do not contest the examiner's assertion that it would have been obvious to apply the verification system of Hollander to "islands" in view of Guruswamy. The admitted prior art in appellants' Figs. 2A and 2B shows that it was known to apply simulation and verification procedures to islands, so we agree that the device under test in Hollander could be an island.

The examiner's rejection does not specifically point out the correspondence between the claim limitations and Hollander, but merely provides a general description of Hollander followed by citations to Hollander. However, the only limitations argued are "wherein the checker further determines the desired output based upon the input" in claim 1 and "the checking step (b) further including the step of (b1) determining the desired output based upon an input" in claims 10 and 17. It is noted that claim 1 recites that the checker "determines" the desired output rather than "generates" the desired output as stated in the arguments. Similarly, claims 10 and 17 recited "determining" rather than "generating." The difference is that "determining the desired

output based upon an input" could be simply deciding which desired output to use for comparison when a particular input is applied, while "generating" implies some kind of calculation of a desired output using the input. Appellants' position is that neither Hollander nor Guruswamy teaches a checker or a checking step that determines the desired output based upon the input. Unfortunately, neither the rejection nor the examiner's response to the arguments addresses these arguments.

Hollander discloses a method and apparatus for functionally verifying an integrated circuit design. The device under test (DUT) 38 in Fig. 1 can be a simulator 36 with an interface 50 (col. 6, lines 59-63). "The invention includes a test generator module 26 for automatically creating a device verification test from a functional description." (Col. 7, lines 12-14.) "A test suite designed with the invention can include any combination of statically and dynamically-generated tests, as well as deterministic and random tests." (Col. 7, lines 22-24.) That is, the test generator generates a suite of tests, where the static, dynamic, deterministic, and random tests (cols. 2-3) include test vectors that are input to the DUT (col. 1, lines 62-67). "The invention permits the user to drive 32 or sample 34 any node in the design. Tests can be generated and expected responses can be computed in advance. The expected results are compared with the DUT output after the test is

simulated. Tests can also be dynamically generated, changing in response to the simulated state of the device." (Col. 8, lines 13-18.) "The runtime environment uses a checker module 30 to verify the design of the DUT. Data checks are performed to verify the relation among different data elements, or to verify data against a high level reference model. The invention can perform any combination of static and dynamic checks. When using both dynamic generation and dynamic checking, the test generator module and the checker can constantly synchronize." (Col. 8, lines 23-30.) We consider the claimed "snooper" to correspond to the "sample 34" in Fig. 1 of Hollander which monitors the output of the DUT; the claimed "generator" to correspond to the "test generator module 26" which generates inputs for the DUT; the claimed "test case" to correspond to the functional description which directs the test generator to create test inputs to the DUT and perform a particular simulation on the DUT; and the claimed "checker" to correspond to the "checker module 30." These limitations are not argued. The only issue argued is whether Hollander discloses or suggests "the checker further determines the desired output based upon the input," as recited in claim 1, or "the checking step (b) further including the step of (b1) determining the desired output based upon an input," as recited in claims 10 and 17.

Initially, we note that the checker in Hollander (and, indeed, any checker by definition) must have at least the "output provided by the island during simulation," i.e., the actual output, and the "desired output" in order to perform a comparison between the two. The "desired output" is broadly "determined" by the checker since it must know what the desired output is for comparison. "Determine" does not imply calculating from an input. The "desired output" is necessarily "based upon the input." One interpretation of "the checker further determines the desired output based upon the input" in claim 1 and "the checking step (b) further including the step of (b1) determining the desired output based upon an input" in claims 10 and 17 is that "determines the desired output" and "determining the desired output" are met by the fact that the checker knows the desired output where the "desired output" being "based upon the input" is an inherent characteristic. That is, it seems that any checker has to have some way to determine what the desired output is so that the desired output can be compared to the actual output, and the desired output is inherently based upon the input.

A second interpretation is that the input is given to the checker, which determines the desired output from the input; this appears to be the intended meaning as evidenced by the specification, page 12, lines 18-19, although it is not the only meaning. The input could be used to index a lookup table of

pre-calculated desired results or the desired output could be calculated from the input. Hollander discloses that "[t]ests can be generated and expected responses can be computed in advance" (col. 8, lines 14-15), where an "expected response" corresponds to the claimed "desired output" and, naturally, must be computed based on a known input. Hollander also describes in the background that "[c]hecking is done by comparing results or behavior with the expected results that are either concluded by the designer or computed by a predictor simulator" (col. 1, lines 53-55). Hollander also states that "[w]hen using both dynamic generation and dynamic checking, the test generator module and the checker can constantly synchronize" (col. 8, lines 28-30), which teaches synchronization of dynamic inputs with checking of expected responses to those inputs. Therefore, some component in Hollander "determines the desired output based upon the input" or performs the step of "determining the desired output based upon an input," as claimed, but Hollander does not state which component it is. It would have been obvious that the predictor simulator which computes the expected result (desired output) based upon an input, or the structure or programming that provides the pre-computed expected response for a certain input, or the part of the dynamic generation and dynamic checking which synchronizes input to expected results, can be considered part of the checker or checking step since this is a mere matter of

labeling a function under one named element or step rather than another. This is especially true for the method step of "the checking step (b) further including the step of (b1) determining the desired output based upon an input," as recited in claims 10 and 17, because a method has no implied structure--it is merely a matter of labeling the determining step, wherever it takes place in Hollander, as part of a checking step. In other words, the determining function is taught in Hollander and the only difference is how the function is labeled as part of a checker component or a checking step. Stating that the determining function is part of the checker instead of some other component, such as the test generator, does not require any modification of Hollander: it is only a matter of labeling or terminology. Accordingly, we will sustain the rejection of claims 1-23.

We note that appellants should also consider whether the claim language is sufficient to define over the admitted prior art (APA) of Figs. 2A-2D. The APA discloses performing the simulation phase of an integrated circuit having an island including an interface using a model and a test case. "[T]he conventional test cases and models provided data to each of the islands ... and check the output of each of the islands." (Specification, p. 2, line 22 to p. 3, line 1.) "[T]he conventional test cases ... include sufficient intelligence to control the conventional models ... during testing. For example,

a conventional test case ... may tell a conventional model ... to input data to an island ..., wait for an output from the island ..., check the output received from the island ... against an expected output, and flag an error if the output does not match the expected output." (Specification, p. 4, lines 8-13.)

Therefore, the conventional model performs the function of "a generator coupled with the island for providing an input to the island during simulation," "a snoopers coupled with the interface for monitoring the interface and obtaining an output provided by the island during simulation," and "a checker, coupled with the interface, for checking the output to determine whether the output is a desired output." Appellants acknowledge that "[t]he snoopers, checker and generator replace the conventional model" (specification, p. 10, line 21). The conventional test case is coupled to and controls the conventional model during testing, so it is "at least one test case coupled with the generator for directing the generator." The conventional model (the generator/snoopers/checker) acts to "check the output received from the island ... against an expected output" and, thus, there is structure or programming somewhere that "determines the desired output based upon the input" where the "desired output" corresponds to the "expected output." That is, the functions are taught in the admitted prior art and the only difference is how the functions are divided among arbitrary component labels.

Appeal No. 2004-1640
Application 09/409,940

Merely dividing the conventional model in the prior art into snooper, checker, and generator functions appears to be a matter of terminology or labeling, not structure or programming. Since the conventional model is controlled by the conventional test case, it appears that the generator part of the conventional model performs the function of "the generator further includes intelligence to provide the input to the island based only upon data and a request provided by the at least one test case to the generator, the request requesting that the generator perform a particular simulation on the island."

Appeal No. 2004-1640
Application 09/409,940

CONCLUSION

The rejection of claims 1-23 under 35 U.S.C. § 112, first paragraph, lack of enablement is reversed.

The rejection of claims 1-23 under § 103(a) is sustained.

No time period for taking any subsequent action in connection with this appeal may be extended under 37 CFR § 1.136(a).

AFFIRMED

KENNETH W. HAIRSTON
Administrative Patent Judge

Jerry Smith
JERRY SMITH
Administrative Patent Judge

LEE E. BARRETT
Administrative Patent Judge

BOARD OF PATENT
APPEALS
AND
INTERFERENCES

Appeal No. 2004-1640
Application 09/409,940

SAWYER LAW GROUP LLP
P.O. BOX 51418
PALO ALTO, CA 94303